

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

Приладобудівний

(повна назва факультету)

Приладобудування

(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_ Гераїмчук М.Д.  
(підпис) (ініціали, прізвище)

“ \_\_\_\_ ” \_\_\_\_\_ р.

**Дипломний проект**

на здобуття ступеня бакалавра

з напрямку підготовки 6.051003 Приладобудування

на тему: Побудова портативних приладів добового моніторингу стану  
фізіологічних систем організму людини

Виконав (-ла): студент (-ка) \_4\_ курсу, групи \_\_ПІ-51\_\_  
(шифр групи)

Євстратов Дмитро Юрійович  
(прізвище, ім'я, по батькові)

(підпис)

Керівник д.т.н, професор, Гераїмчук Михайло Дем'янович \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант технологічний Антонюк Віктор Степанович \_\_\_\_\_  
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали)

(підпис)

Рецензент д.т.н, професор, Шевченко Вадим Володимирович \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цьому дипломному  
проекті немає запозичень з праць інших  
авторів без відповідних посилань.

Студент \_\_\_\_\_

Київ – 2019 року

# **Пояснювальна записка до дипломного проекту**

на тему: Побудова портативних приладів добового моніторингу стану  
фізіологічних

систем організму людини

---

Київ – 2019 року

**Національний технічний університет України  
«Київський політехнічний інститут  
імені Ігоря Сікорського»**

Факультет Приладобудівний  
(повна назва)

Кафедра Приладобудування  
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки 6.051003 Приладобудування  
(код і назва)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
\_\_\_\_\_ Гераїмчук М.Д.  
(підпис) (ініціали, прізвище)  
«\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на дипломний проект студенту**  
**Євстратову Дмитру Юрійовичу**  
(прізвище, ім'я, по батькові)

1. Тема проекту Побудова портативних приладів добового моніторингу стану фізіологічних систем організму людини \_\_\_\_\_

керівник проекту д.т.н, професор, Гераїмчук М.Д. \_\_\_\_\_  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від « 27 » 05 2019 р. № 1384-с

2. Термін подання студентом проекту \_\_\_\_\_

3. Вихідні дані до проекту Мікроконтролер Arduino \_\_\_\_\_

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити) Розглянути технічні можливості платформи Arduino, Вивчення технології Bluetooth low energy, робота з протоколом фітнес-трекера, розробка програмного продукту, тестування, Розробити програмний інтерфейс для отримання даних з Arduino,

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо) 3 кресленника А1 \_\_\_\_\_

6. Консультанти розділів проекту (роботи)\*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Технологічний	Антонюк В.С. д.т.н. професор		

7. Дата видачі завдання 19.05.2019 \_\_\_\_\_

Календарний план

№ з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Отримання завдання	19.05.2019	
2	Збір інформації	20.05.2019	
3	Вивчення варіантів реалізації та вибір варіанту для розробки	23.05.2019	
4	Розробка схеми аналогової частини приладу	25.05.2019	
5	Розробка програмного забезпечення	27.05.2019	
6	Розробка опису приладу	29.05.2019	
7	Оформлення дипломної роботи	04.05.2019	
8	Отримання допуску до захисту та подача роботи в ДЕК	15.05.2019	

Студент

\_\_\_\_\_  
(підпис)

\_\_\_\_\_  
(ініціали, прізвище)

Керівник проекту

\_\_\_\_\_  
(підпис)

\_\_\_\_\_  
(ініціали, прізвище)

\_\_\_\_\_  
\* Консультантом не може бути зазначено керівника дипломного проекту (роботи)

# Реферат

Дипломна робота складає 59 сторінок, 24 рисунки, 5 таблиць, 3 кресленика А1.

Метою проекту є розробка портативного багатоканального мікропроцесорного кардіографа. Попередньо розглянуті походження і різні методи знімання біопотенціалів серця. Показано особливості режимів роботи кардіографа: автономного та комплексного.

На підставі пред'явлених технічних вимог проводиться розробка структурної, функціональної та принципової схем блоків приладу, проводиться розрахунок похибок вимірювання амплітуди і тимчасових інтервалів кардіосигналу.

## Abstract

Thesis consists of 83 pages, 24 figures, 5 tables, 3 drawings A1.

The aim of the project is to develop a portable multi-channel microprocessor cardiograph. Initially considered the origin and various methods of removal of the biopotentials of the heart. The features of the cardiograph operation modes are shown: autonomous and complex.

On the basis of the submitted technical requirements, the structural, functional, and schematic diagrams of the instrument blocks are developed, the errors of measuring the amplitude and time intervals of the cardio signal are calculated.

# 3MICT

ЗМІСТ.....	7
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	9
ВСТУП.....	10
1 ПОХОДЖЕННЯ ТА РЕЄСТРАЦІЯ ЕЛЕКТРОКАРДІОГРАМИ .....	12
1.1 Величина і напрям зубців електрокардіограми .....	14
1.2 Технічні вимоги до розроблень кардіографа.....	15
2 ПЛАТФОРМА ARDUINO .....	16
2.1 Апаратна частина платформи Arduino .....	17
2.2 Версії різних платформ Arduino .....	18
2.3 Arduino Leonardo .....	19
2.3.1 Загальні відомості.....	20
2.3.2 Характеристики .....	20
2.3.3 Живлення.....	22
4 ПІДСИЛЮВАЧ СИГНАЛУ .....	23
<b>5 ОЦІНКА СКЛАДАЛЬНОЇ ОДИНИЦІ НА ТЕХНОЛОГІЧНІСТЬ.....</b>	<b>27</b>
5.1 Розрахунок пристроїв захисту та вхідних підсилювачів.....	29
5.2 Розрахунок і вибір елементів пристрою гальванічної розв'язки.....	32
5.3 Розрахунок похибки вимірювання амплітуди кардіосигналу.....	34
5.4 Розрахунок похибки аналого-цифрової частини ПММК.....	35
5.5 Розрахунок напруги шуму аналого-цифрової частини ПММК .....	35
6 ВИКОРИСТОВУВАНІ МОВИ ПРОГРАМУВАННЯ І ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ .....	41
6.1 Мова програмування C #.....	42
6.2 Опис мови XAML.....	43
6.3 Середовище розробки Visual Studio 2015 .....	43
6.4 Платформа Windows Phone 8.1 .....	44

					ДП. ПІ 51.00.000 ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Євстратов			Портативний кардіограф  Пояснювальна записка	Літ.	Арк.	Акрушіє	
Перевір.									
Реценз.									
Н. Контр.									
Затверд.		Гераїмчук							

6.4.1 Архітектура платформи.....	45
6.4.2 Безпека Windows Phone .....	46
6.4.3 Ядро .....	47
6.5 Інструменти для розробки.....	48
6.6 Інтерфейс користувача на платформі .....	49
6.7 Структура проекту Windows Phone .....	49
6.8 Життєвий цикл додатку .....	51
7 BLUETOOTH .....	51
7.1 Bluetooth low energy .....	52
7.2 Архітектура програми .....	53
ВИСНОВКИ.....	55
ПЕРЕЛІК ПОСИЛАНЬ .....	56



## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ОП - операційний підсилювач

KIX – кінцева імпульсна характеристика

ЕКГ - електрокардіограма

POE - Power over Ethernet

LGPL - Lesser General Public License

АЦП - аналого-цифровий перетворювач

HIX – нескінченна імпульсна характеристика

ШПФ - швидке перетворення фур'є

ЕОМ - електронна обчислювальна машина

JSSC - Java Simple Serial Connector

USB - Universal Serial Bus

					ДП. ПІ 51.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дат		

## ВСТУП

З ростом і розвитком науки і техніки, а також, і медицини, виникає необхідність в розробці більш новітніх і зручних в обігу медичних приладів. З інтеграцією в усі сфери людської діяльності інформаційних та, зокрема, комп'ютерних технологій виникла необхідність створення нових засобів медичної діагностики із застосуванням ЕОМ. Це значно полегшує роботу лікаря, і суттєво економить час пацієнта. В даний час люди більшою мірою схильні до різних захворювань, в тому числі і серцево-судинним, які за статистикою є причиною більшості смертей. Тому діагностика серцево-судинної системи є однією з найважливіших задач медицини, а інформаційні технології покликані допомагати цьому шляхом впровадження новітніх розробок в медичні прилади. Одним з таких приладів є, що розробляється портативний багатоканальний мікропроцесорний кардіограф [2]. Цей гаджет може широко застосовуватися в лікувальній та діагностичній медицині. Функціональним елементом серця слугує м'язове волокно - ланцюжок з клітин міокарда, сполучених кінець в кінець і вкладених в загальну оболонку. В залежності від морфологічних і функціональних особливостей в серці існують два типи волокон: типові - волокна робочого міокарда передсердь і шлуночків, що складають основну масу серця і забезпечують його нагнітальну функцію; атипові - волокна водія ритму і провідної системи, що відповідають за генерацію збудження і проведення його до клітин робочого міокарду. Серцевому м'язу притаманні особливі властивості, завдяки яким здійснюється її складна діяльність. Це - функції автоматизму, збудливості, провідності і скорочення. М'яз серця (міокард) належить

					ДП. ПІ 51.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дат		

до збудливих тканин. Це означає, що волокна міокарда мають потенціал спокою, відповідають на надпорогові стимули генерацією потенціалів дії і здатні проводити ці потенціали без загасання. Потенціал спокою волокон міокарда коливається в широких межах в залежності від функціонального характеру волокна. У скоротливих, типових волокон він дорівнює в середньому  $-90\text{мВ}$  [2], тоді як у атипівих клітин синусового вузла становить не більше  $-60\text{мВ}$ . М'язова тканина передсердь і шлуночків поводитья як функціональний синцитій: збудження, що виникає в будь-якому з цих відділів, охоплює всі без винятку збуджені волокна [2]. Завдяки цій функції серце підкоряється формулі все або нічого: на роздратування воно або реагує збудженням всіх волокон, або не реагує зовсім. Ритмічні скорочення серця виникають автоматично, без будь-якого впливу на нього будь-яких зовнішніх спонукальних чинників – подразників [2]. Однак скорочення типової мускулатури серця може відбутися лише в тому випадку, якщо попередньо в якій-небудь ділянці цього органу виникне електричний імпульс збудження. Завдяки мимовільному ритмічному порушенню і відбувається подальше ритмічне скорочення міокарда. Ритмічні скорочення серця з'являються під впливом імпульсів, які зароджуються в ньому. Ця властивість називається автоматизмом. У здоровому серці імпульси зароджуються тільки в синусному вузлі. В результаті внутрішніх обмінних процесів, суть яких до сих пір остаточно не ясна, в цьому шматочку тканини через визначені проміжки часу відбувається невеликий електричний розряд, що запускає скорочення серця. Таким чином, синусний вузол служить свого роду "запальною свічкою". Він є водієм ритму, автоматичним центром першого порядку. Він же останнім перестає функціонувати при зупинці серця. Зародження

імпульсу у водії ритму приводить до поширення збудження по всьому міокарду. У активність втягується вся маса скорочувальних волокон серця. У кожному з них реєструються потенціали дії. Збуджена ділянка волокна завжди заряджена електронегативно по відношенню до сусідньо збудженій ділянці. Те ж саме стосується і цілого серця. Хвиля деполяризації охоплює серцевий м'яз не зовсім одночасно, а в певній послідовності, швидко переходячи з одних груп м'язових волокон на інші. Тому в працюючому серці завжди є порушені і ще не порушені або вже завершили збудження ділянки, між якими виникає різниця потенціалів, яка народжує електрорушійну силу (ЕРС). Зареєструвати ЕРС серця можна, доклавши безпосередньо до його поверхні два відвідних електрода, з'єднаних з осцилографом. При цьому вдається записати електрограму серця, що представляє собою алгебраїчну суму потенціалів дії різних зон міокарда. Однак цей метод використовується, головним чином, в експериментах на тваринах. Для людини, з цілком зрозумілих причин, він не прийнятний. Силкові лінії електричного поля серця поширюються по всьому тілу і можуть бути зареєстровані з будь-яких точок його шкірної поверхні. Інакше кажучи, між будь-якими двома точками поверхні тіла можна зареєструвати різницю потенціалів, що є відображенням електричної діяльності порушеного міокарда. Крива змін цієї різниці потенціалів в часі називається електрокардіограмою, а метод реєстрації - електрокардіографією.

## 1 ПОХОДЖЕННЯ ТА РЕЄСТРАЦІЯ ЕЛЕКТРОКАРДІОГРАМИ

При вивченні генезу електрокардіограми необхідно використовувати такі факти: загальне електричне поле серця утворюється в результаті складання

					<i>ДП. ПІ 51.00.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		12

полів численних окремих волокон серця; кожне збуджене волокно являє собою диполь, що володіє елементарним дипольним вектором певної величини і напрямки; інтегральний вектор в кожен момент процесу збудження являє собою результуючу окремих векторів; величина потенціалу, що вимірюється в точці, віддаленій від джерела, залежить головним чином від величини інтегрального вектора і від кута між напрямком цього вектора і віссю відведення.

Спеціалізована система дуже швидко проводить збудження до різних відділах шлуночків. Внаслідок цього розмір ділянки міокарда, активується одним волокном Пуркінє (по цій ділянці хвиля збудження рухається безперервно), відносно невелика (близько 1 см). Ця відстань називається довжиною вільного пробігу. Довжину хвилі збудження можна розрахувати, помноживши швидкість проведення (близько 1 м / с) на тривалість збудження (близько 0,3 с); вона дорівнює 0,3 м. З цього випливає, що в кожен момент циклу збудження в серці можуть існувати лише невеликі ділянки хвилі збудження.

У міру того як хвиля збудження поширюється по волокну міокарда на ділянці, відповідному довжині вільного пробігу, створюється градієнт напруги ( $dV / dx$ ), величина якого в кожен момент залежить від фази збудження. В області фронту хвилі є крутий градієнт величиною 120 мВ (відповідної амплітуді потенціалу дії) на ділянці довжиною близько 2 мм (градієнт напруги дорівнює 600 мВ / см). Навпаки, під час фази реполяризації виникають набагато менші градієнти напруги, спрямовані в зворотну сторону. У першому наближенні волокно міокарда поводить себе в фізичному відношенні як змінний диполь, що характеризується певною величиною і напрямком. За визначенням дипольний вектор направлений від мінуса до плюса, тобто від порушеної ділянки до збудженої (збуджена

ділянка зовні заряджена негативно по відношенню до збудженої). Дипольний вектор переднього фронту хвилі збудження можна назвати вектором деполяризації, а вектор її заднього фронту, спрямований у протилежний бік, вектором реполяризації.

У кожен момент в процесі збудження серця окремі вектори сумуються і утворюють інтегральний вектор. Він будується так само, як результуюча двох сил за правилом паралелограма; при цьому виходячи з двох векторів будується третій. У середині стінки серця велика частина векторів (за підрахунками до 90%) діє у взаємопротилежних напрямках і нейтралізує один одного.

### 1.1 Величина і напрям зубців електрокардіограми

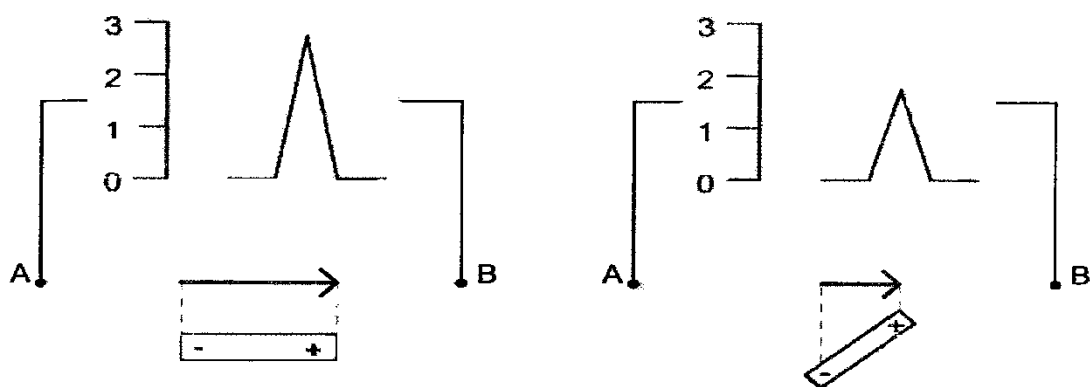


Рисунок 1.1- Залежність величини зубця від електричного поля диполя

Для того, щоб виявити зв'язок між орієнтацією вектора серця і полярністю зубців електрокардіограми, необхідно розглянути електричне поле навколо диполя, поміщеного в однорідно провідне середовище. Точки цього поля, що володіють однаковими потенціалами, утворюють так звані ізопотенціальні лінії. З малюнка 1.1 видно, що відстань між точками А і Б залежить перш за все від кута між віссю диполя і віссю відведення і

дорівнює проекції інтегрального вектора на вісь відведення. Якщо напрямок відведення збігається з напрямком інтегрального вектора, величина реєструється різниці потенціалів максимальна. Якщо ж ці напрямки взаємно перпендикулярні, різниця потенціалів дорівнює нулю. В принципі це правило можна перенести і на електрокардіограму людини, хоча на практиці в цьому випадку картина значно складніше. Це пов'язано з тим, що, по-перше, тіло людини не є електрично однорідним середовищем, по-друге, серце розташоване не в серці сферичного провідника. У зв'язку з цим електричне поле серця на поверхні тіла спотворюється.

## 1.2 Технічні вимоги до розроблень кардіографа

Основні технічні вимоги проектованого електрокардіографа повинні задовольняти ГОСТ 19687-89 "Прилади для вимірювання біоелектричних потенціалів серця. Загальні технічні вимоги та методи випробувань." Основні параметри електрокардіографа повинні відповідати наведеним нижче:

Діапазон входних напруг, мВ	± 250
Максимальна приведена похибка вимірювання амплітуди кардіосигналу, %	0.5
Максимальна абсолютна похибка вимірювання тимчасового інтервалу в кардіосигналі не більше, мс	5
Чутливість, мкВ	3
Вхідний імпеданс, Мом	5
Коефіцієнт ослаблення синфазних сигналів, не менше	100000
Напруга внутрішніх шумів, поріведенних до входу, мкВ, не більше	2
Нерівномірність амплітудно-частотної характеристики в діапазоні частот	90...105 70...105

від 0.5 до 50 Гц, %	1000
---------------------	------

АЦП має синхронний послідовний інтерфейс для обміну даними з процесором. В інтерфейсі використовуються наступні сигнали: SCLK - синхроімпульсів, які забезпечують синхронізацію біт; RFS - синхронізація прийому кадру; TFS - синхронізація передачі кадру; DR - прийом даних; DT - передача даних. Також є сигнали каскадирування, що дозволяють об'єднувати до восьми мікросхем даного типу. Швидкість передачі з синхронного послідовного порту може досягати 8МБ / с. Так як кожна мікросхема містить чотири АЦП нам необхідно три таких мікросхеми. Для управління ними необхідно три сигнали: RESET - запуск перетворення вхідного сигналу; DRDY - завершення перетворення вхідного сигналу; CLKIN - тактова частота для синхронізації АЦП.

## 2 ПЛАТФОРМА ARDUINO

Цифрова частина приладу є мікроконтролером Arduino Leonardo.

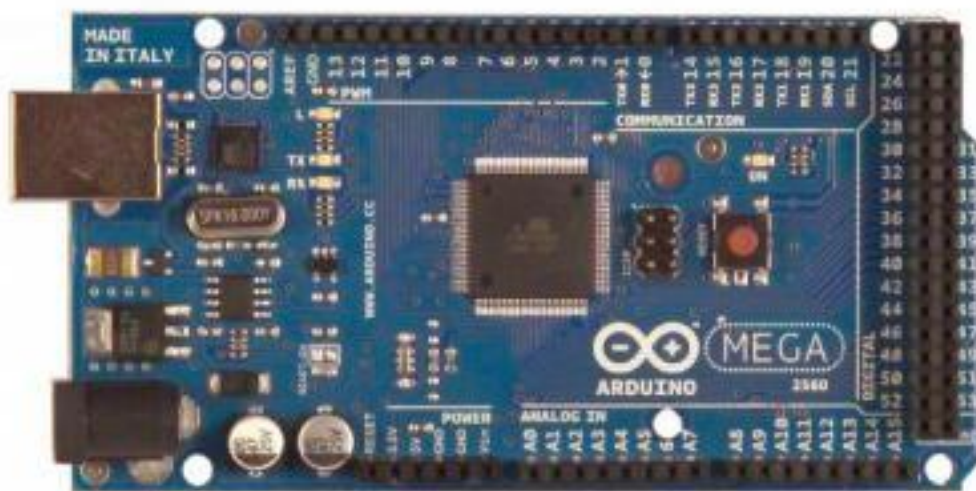


Рисунок 2.1 - Мікроконтролер Arduino Mega

Змн.	Арк.	№ докум.	Підпис	Дат

ДП. ПІ 51.00.000 ПЗ

Арк.

16



Arduino – сучасний електронний конструктор і зручна платформа оперативної розробки електронних приладів для новачків і професіоналів. Дана платформа користується великим попитом в усьому світі завдяки зручним умовам роботи з нею у якості програмування, а також, вільній архітектурі і програмному коду. Прилад програмується через USB без необхідності використання програма торів [1].

Arduino дозволяє ПК вийти за рамки віртуального світу в фізичний і взаємодіяти з ним. Пристрої на базі Arduino можуть отримувати данні про навколишній простір за допомогою різних датчиків, а також, можуть контролювати різними виконавчими пристроями [1]. Мікроконтролер на платі запрограмований на мові Arduino (на основі мови підключення) і середовищі програмування Arduino (на основі середовища обробки) [1]. Конструкції пристроїв на основі Arduino можуть працювати незалежно або взаємодіяти з програмним забезпеченням на комп'ютері (наприклад, Flash, Processing, MaxMSP). Платежі можуть бути зібрані користувачем самостійно або придбані в колекції [1]. Програмне забезпечення можна завантажити безкоштовно. Схематичні креслення (файли CAD) є загальнодоступними, користувачі можуть використовувати їх за власним розсудом. У 2006 році Arduino була визнана категорією цифрових спільнот на Ars Electronica Prix [1].

## 2.1 Апаратна частина платформи Arduino

Існує декілька версій платформ Arduino. Версія для Leonardo базується на мікроконтролері ATmega32u4. Незважаючи на те, що попередня версія Duemilanove базується на мікроконтролері Atmel ATmega328. Версія платформи Diecimila і перша робоча машина були створені на базі Atmel ATmega168, більш ранні версії використовували ATmega8. Arduino Mega2560, в свою чергу, побудований на мікроконтролері ATmega2560..[1]

					ДП. ПІ 51.00.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дат		

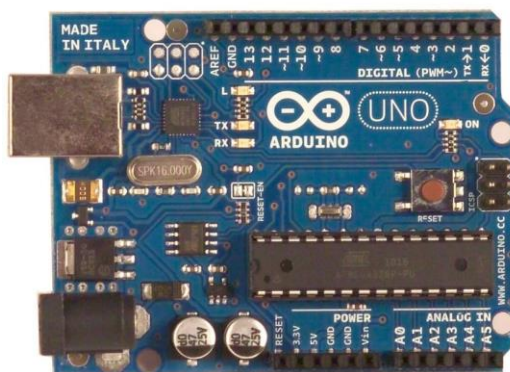


Рисунок 2.2 – Arduino Uno

## 2.2 Версії різних платформ Arduino

Нижче вказані основні версії плат Arduino [1]:

**Leonardo** - найновіша версія платформи Arduino на ATmega32u4.

**Due** - плата на базі 32bit Cortex-M3 ARM SAM3U4E.

**Yun** - Плата, з інтегрованою підтримкою WiFi на основі ATmega32u4, а також, Atheros AR9331

**Uno** - найвідоміша версія звичайної платформи Arduino USB. Дана платформа може бути покращена платами розширення, таких, як плати юзерів з різним функціоналом [1].

**Micro** - компактне рішення на базі ATmega32u4.

**Arduino Ethernet** - контролер з інтегрованою підтримкою функціонування по мережі за участі модуля POE.

**Duemilanove** - є передостанньою версією стартової платформи Arduino USB. Після підключення вона здатна до експлуатації [1].

**Diecimila** - стара версія платформи Arduino USB.

**Nano** - більш зручна платформа, використовувана як макет. Nano підключається до ПК за допомогою кабеля USB Mini-B.

**Mega2560** - нова версія плати серії Mega [1]. Створена на базі Atmega2560 і з використанням чіпа ATmega8U2 для послідовного

під'єднання через USB порт.

**Mega** - більш стара версія серії Mega на базі Atmega1280 [1].

**Arduino BT** - платформа з модулем «Блютуз» для бездротового зв'язку.

### Плати розширення

Плати розширення - це плати, які розширюють пристосованість Arduino для контролю різними приладами, отримання даних і іншими функціями.

Плата WiFi слугує для сполучення з бездротовими мережами [1].

Xbee Shield виконує за допомогою модуля Maxstream Xbee Zigbee wireless зв'язок одночасно між багатьма приладами Arduino в радіусі 35 метрів (в приміщенні) і 90 метрів (поза приміщенням) [1].

Motor Shield виконує функцію контролю двигунами постійного струму і зчитування датчиків положення.

Ethernet Shield забезпечує під'єднання до інтернету.

## 2.3 Arduino Leonardo



Рисунок 2.3 – Arduino Leonardo

### 2.3.1 Загальні відомості

Платформа Ардуіно дозволяє створювати найрізноманітніші автономні пристрої, але ось при підключенні до комп'ютера традиційних плат Uno, Nano, Mega неможливо було змусити їх працювати як звичайну периферію - наприклад, мишки або клавіатури. Не можна просто підключити звичайний джойстик через Ардуіно уно до комп'ютера і керувати ним, наприклад, курсором - доведеться встановлювати додаток на самому комп'ютері, яке буде інтерпретувати коди з послідовного порту і встановлювати курсор в потрібне місце.

Плата Arduino Leonardo істотно розширює можливості для підключення Arduino до комп'ютера. Вона розглядається комп'ютером як стандартний HID-пристрій і може безпосередньо надсилати потрібні керуючі сигнали. Не можна сказати, що ця плата може замінити Uno, але володіючи унікальними властивостями, вона може стати основною нових дуже цікавих і незвичайних проектів. В основі Leonardo лежить досить просунутий мікроконтролер ATmega32u4. Сама плата досить сильно схожа на Uno, на ній можна знайти два десятка входів і виходів, роз'єм micro-USB, кнопка для перезавантаження, а також роз'єми ICSP і силовий вхід. Живлення плати стандартне для Ардуіно, рекомендованої напруги 7-12 вольт.

Кількість цифрових входів і виходів 20, а аналогових - 12. До звичайних аналоговим пінам A0-A5 тут додаються ще Піни 4,6,8, 9,10, 12, підключені до АЦП. Вбудована flash-пам'ять має об'єм 32 КБ, з яких чотири призначені завантажувачу. ОЗУ пристрою 2,5 КБ, а тактова частота дорівнює 16 МГц.

### 2.3.2 Характеристики

Мікроконтролер	ATmega32u4
----------------	------------

					ДП. ПІ 51.00.000 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дат		

Робоча напруга	Робоча напруга
Вхідна напруга (рекомендована)	7-12 В
Вхідна напруга (граничне)	6-20 В
Цифрові Входи / Виходи	20 (7 з яких можуть використовуватися як виходи ШІМ)
Аналогові канали	12
Постійний струм через вхід / вихід	40 мА
Постійний струм для виведення 3.3 В	50 мА
Флеш-пам'ять	32 кБ (АТmega32u4) з яких 4 кБ використовуються для завантажувача
ОЗУ	2 кБ (АТmega32u4) EEPROM 1 кБ (АТmega32u4)
Тактова частота	16 МГц

Таблиця 2.1 – Характеристики Arduino Leonardo

### 2.3.3 Живлення

#### Живлення Arduino через роз'єм DC (5,5мм / 2,1 мм)

У повнорозмірних плат Arduino (Duemilanove, Uno, Mega, Leonardo і т. П.) Є лінійний регулятор напруги і роз'єм підключення блоку живлення 5,5мм / 2,1 мм (зовнішній / внутрішній діаметр). До нього можна підключати джерело живлення постійного струму з напругою від 9 до 12 Вольт [17].

Якщо у вас є блок живлення з напругою менше 7 В, то плата, можливо, буде працювати, але на портах введення-виведення плати буде вже не 5 Вольт, а менше і ваша схема буде працювати менш передбачувано [17]. З іншого боку, напруга вище рекомендованого змусить регулятор напруги на платі працювати в режимі перевантаження, перегріватися і виділяти надто багато тепла. Золота середина - 9 ... 12 Вольт - дозволить регулятору напруги розсіювати виділяється на радіаторі тепло без перевантажень і подавати на мікроконтролер необхідну напругу. Слід зазначити, що на виведення Vin плати буде не стабілізовану напругу 5 В, а напруга на вході в плату. Це зручно, наприклад, у разі якщо у вас є реле з обмоткою на 12 В і у ваш блок живлення на 12 В - просто подавайте напругу з Vin [17].

Споживаний платою струм залежить від того, які частини мікроконтролера працюють в даний момент і яке навантаження підключена до плати. Чим більший струм може забезпечити ваш блок живлення, тим краще, це означає, що при великому навантаженні на нього він не буде перегріватися.

Мінімальна вимога - 250 мА, що дозволять запустити тільки лише одну плату Arduino без підключених до неї датчиків або активаторів. Краще - джерело живлення на струм від 0,5 А до 2 А [17].

Ось деякі обмеження по струму для різних пристроїв:

					ДП. ПІ 51.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дат		

порти USB мають автоматичний запобіжник на 500 мА, який повинен відключити порт при його перевантаженні. Будь-який пристрій, який споживає струм більше 500 мА, підключений до порту, буде працювати непередбачувано чи не працювати зовсім. Максимальний струм для порту вводи-виводи мікроконтролера ATmega складаємо 40 мА (рекомендується не більше 35 мА) при загальному струмі споживання мікро контролером не більше 200 мА [17]. Максимальний струм, який може витримати порт 5V плати Arduino, складає 0,8 А, проте рекомендується допускати не більше 500 мА. Порт Vin плати Arduino міг би витримати такий же струм, як і джерело живлення, але його струм обмежений захисним діодом цього ланцюга, а він розрахований та струм 1А. Деяка периферія (такі як двигуни, реле, нагрівальні елементи і т.п.) можуть споживати дуже великий струм, так що вам слід уважно читати опис компонентів і забезпечувати необхідний струм. Фактично, пристрій, яке споживає струм більше 0,5 А, має бути живиться від свого джерела [17].

## 4 ПІДСИЛЮВАЧ СИГНАЛУ

Для аналогової частини був обраний інструментальний підсилювач AD620 [12] тому що в документації цього підсилювача є готове рівняння для реєстрації ЕКГ.

Електрокардіограф складається з двох частин - аналогової та цифрової  
Рисунок 4.1. Аналогова частина являє собою підсилювач з коефіцієнтом посилення близько 1000 - він підсилює зміни в часі різницю потенціалів на тілі людини (ЕКГ) приблизно в 1000 разів. Далі посилений сигнал надходить на цифрову частину, де сигнал оцифровується і подається в цифровому вигляді на комп'ютер.

					ДП. ПІ 51.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дат		



Рисунок 4.1- Спрощена схема пристрою

В результаті комп'ютер за допомогою спеціальної програми розшифровує отриманий від цифрової частини цифровий сигнал і показує на екрані монітора графік електрокардіограми (ЕКГ).

Для спрощення виготовлення приладу в якості цифрової була вибрана відладна плата Arduino. У підсумку розробка цифрової частини зводиться лише до налаштування плати, написання відповідної програми (скетчу) - ця програма "пояснює" Arduino з якого роз'єму плата буде читати сигнал (роз'єм A0), щоб оцифрувати і в якому форматі цей оцифрований сигнал буде передаватися на комп'ютер.

Аналогова частина була побудована на базі інструментального операційного підсилювача AD620. У технічній документації на AD620 електрокардіограф будується за наступною схемою Рисунок 4.2:

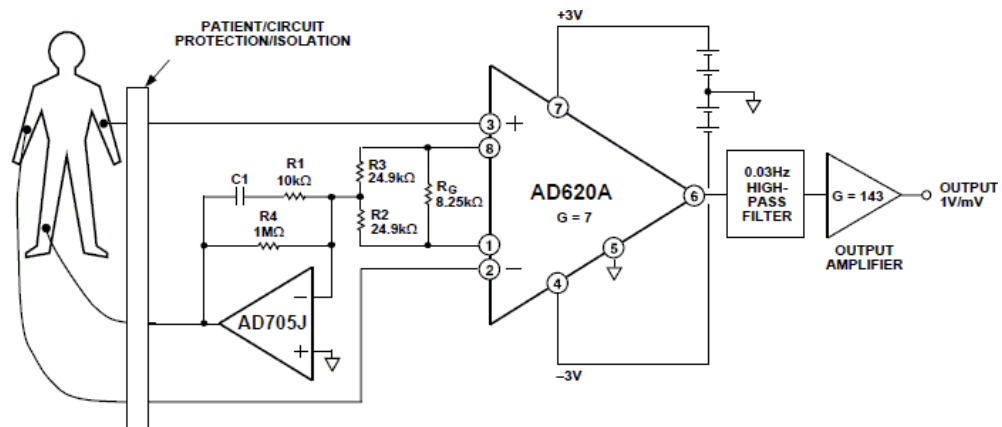


Рисунок 4.3- Принципова схема пристрою



Схему можна спростити наступним чином:

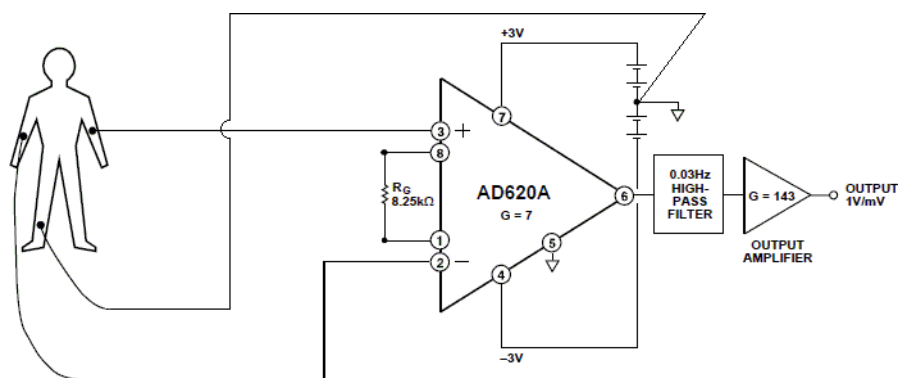


Рисунок 4.4-Спрощена принципова схема пристрою

Схема була спрощена так як "драйвер правої ноги" (right leg driver - "референтний електрод") на основі AD705J не потрібен. Він призначений для боротьби з мережевою наводкою в 50 Гц, яка зазвичай виникає, якщо прилад гальванічно з'єднаний з мережею в 50Гц. Так як наш прилад з мережею НЕ з'єднаний і живиться від батарейки (крони) в 9В, тому схема "драйвера пра- виття ноги" на основі підсилювачів не несе в собі необхідність.

Як "драйвера правої ноги" у нас буде просто нульовий про від. Різниця потенціалів між тілом людини і нульовим потенціалом схеми може бути досить великою. В результаті може вийти так, що по потенціали, що знімаються з тіла і подаються на входи підсилювача щодо нульової точки схеми, мають занадто велику відмінність і таким чином виходять за діапазон допустимих величин потенціалів, що подаються на входи ОУ і це незважаючи на те, що різниця потенціалів між плюсовим і мінусовим входом підсилювача в межах мілівольта. Подібне загрожує режимом насичення підсилювача, в якому він не посилює, а просто видає на вихід постійну складову рівну приблизно верхнього або нижнього значенням напруги живлення підсилювача. Щоб цього уникнути потрібно, провести потенціали тіла до необхідного діапазону - до рівня близького потенціалу

нульового проводу схеми. Для цього і служить нульовий провід, підключати до правої ноги людини. На вході підсилювача крім змінюється різниці потенціалів, зумовленої електричною активністю серця (корисний сигнал) з'являється постійна складова, величина якої може досягати 300 мВ як в позитивну, так і в негативну сторону.

Якщо відразу посилювати цей сигнал, то в підсумку вийде величина близько 300В або -300В - що виходить за межі можливостей підсилювача. Щоб цього уникнути схема була побудована з двох інструментальних посилювачів і фільтра високих частот (ФВЧ) між ними. Перший підсилювач AD620 має коефіцієнт посилення 7, далі у посиленого таким чином сигналу за допомогою ФВЧ у вигляді RC-ланцюжка. Далі отриманий сигнал поступає на 2-й підсилювач з коефіцієнтом посилення 140

Рисунок 4.5:

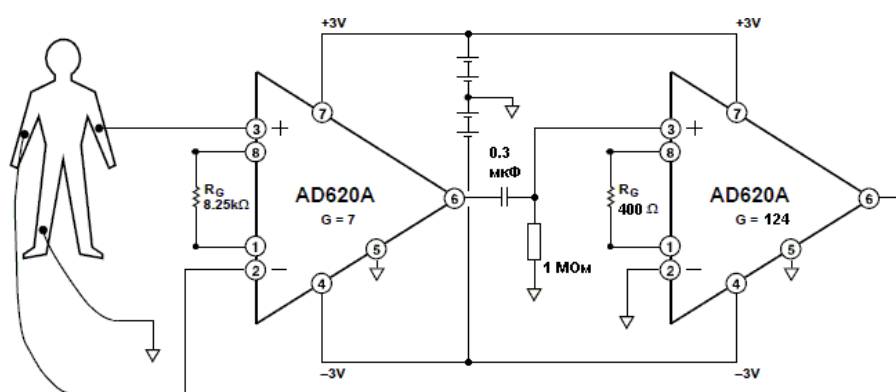


Рисунок 4.5-Принципова схема пристрою

Так як в технічній документації на AD620 написано, що живлення підсилювачів походить від  $\pm 2,5$  В до  $\pm 18$ В було вирішено використовувати схему стабілізації живлення, щоб перетворити живлення від батарейки в  $9\text{В} \pm 2,5$  В.

Принципова схема стабілізатора живлення, побудована в про- програмних середовищі DipTrace Рисунок 4.6.

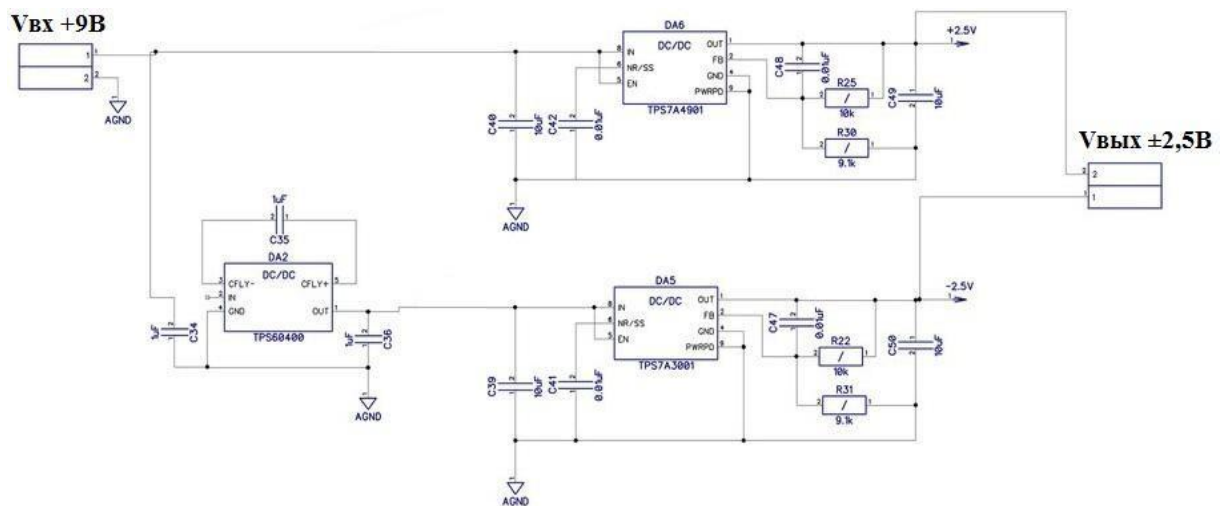


Рисунок 4.6- Принципова схема стабілізатора живлення

У схемі використовувалися наступні мікросхеми:

- Інвертор напруги TPS60400;
- Стабілізатор позитивної напруги TPS7A4901;
- Стабілізатор негативної напруги TPS7A3001.

## 5 ОЦІНКА СКЛАДАЛЬНОЇ ОДИНИЦІ НА ТЕХНОЛОГІЧНІСТЬ

Технологічність конструкції оцінюється кількісними характеристиками, званими показниками технологічності. Розрахункові показники технологічності порівнюють з базовими показниками технологічності. Конструкція вважається технологічною, якщо значення показників технологічності відповідають або перевершують значення базових показників. Для дослідної партії електронних блоків  $K_n = 0.4-0.7$ .

1. Коефіцієнт використання інтегральних мікросхем:

$$K_{\text{испИМС}} = \frac{H_{\text{ИМС}}}{H_{\text{ИМС}} + H_{\text{ЭРЭ}}} = 15 / (15 + 84) = 0.15 \quad (5.1)$$

де НІМС - число інтегральних мікросхем,

Нере - загальне число компонентів

2. Коефіцієнт монтажних з'єднань, здійснюваних автоматизованим і механізованим способом:

$$K_{\text{а.м}} = H_{\text{а.м}} / H_{\text{м}} = 45 / 57 = 0,78$$

3. Коефіцієнт ЕРЕ, підготовка і монтаж яких, здійснюється механізованим способом:

$$K_{\text{мпЭРЭ}} = H_{\text{мпЭРЭ}} / H_{\text{ЭРЭ}} = 45 / 45 = 1 \quad (5.2)$$

4. Коефіцієнт операцій контролю і налаштування здійснюваних автоматизованим і механізованим способом:

$$K_{\text{м.к.н.}} = H_{\text{м.к.н.}} / H_{\text{ЭРЭ}} = 0 \quad (5.3)$$

5. Коефіцієнт монтажу типорозмірів:

$$K_{\text{повЭРЭ}} = 1 - H_{\text{т.ор.ЭРЭ}} / H_{\text{ЭРЭ}} = 1 - 0,07 = 0,93$$

6. Коефіцієнт монтажу оригінальних типорозмірів:

$$K_{\text{повЭРЭ}} = 1 - H_{\text{т.ор.ЭРЭ}} / H_{\text{тЭРЭ}} = 1 - 0 = 1$$

7. Коефіцієнт:

$$K_{\text{ф}} = D_{\text{пр}} / D = 1 / 1 = 1 \quad (5.4)$$

Технологічність виробу оцінюється комплексним показником технологічності, що визначаються на основі базових показників:

$$K = \frac{\sum_{i=1}^n K_i \varphi_i}{\sum_{i=1}^n \varphi_i}, \quad (5.5)$$

де  $K_i$  - розрахунковий базовий показник відповідного класу блоків.

$\varphi_i$  - коефіцієнт ваговій значущості показника;

$i$  - порядковий номер показника в ранжированій послідовності.

$n$  - число базових показників, які визначаються на даній стадії розробки виробу.

$$K = \frac{1 \times 0.78 + 1 \times 1 + 0 \times 0.75 + 0 \times 0.5 + 0.93 \times 0.31 + 1 \times 0.187 + 1 \times 0.11}{0.15 + 0 + 0 + 0 + 0.93 + 1 + 1} = \frac{2.53}{4.71} = 0.53$$

Рівень технологічності розроблюваного виробу при відомому нормативному комплексному показнику згідно ГОСТ 14.202-73 [13], оцінюють відношенням досягнутого комплексного показника до нормативного  $K_n$ . Це відношення має задовольняти умові  $\frac{K}{K_n} > 1$ . Для електронних блоків:

$$K_n = 0.53$$

$$\frac{0.53}{0.4} = 1.34$$

$$1.34 \geq 1$$

Тобто умова технологічності виконується для дослідної партії.

### 5.1 Розрахунок пристроїв захисту та вхідних підсилювачів

Як пристрій захисту застосовуємо діодний обмежувач напруги (ДН), зібраного на елементах: опорі  $R1$  і діодах  $VD1$ ,  $VD2$ . Для завдання рівня обмеження застосовуються джерела напруги зібрані на опорах  $R2$ ,  $R3$  і діодів  $VD1$ ,  $VD2$ .

Вхідний підсилювач представлений у вигляді неінвертуючий підсилювача коефіцієнти посилення, якого задаються за допомогою опорів R4 і R5 і розраховується за формулою:

$$K_u = \frac{R4 + R5}{R5} \quad (5.6)$$

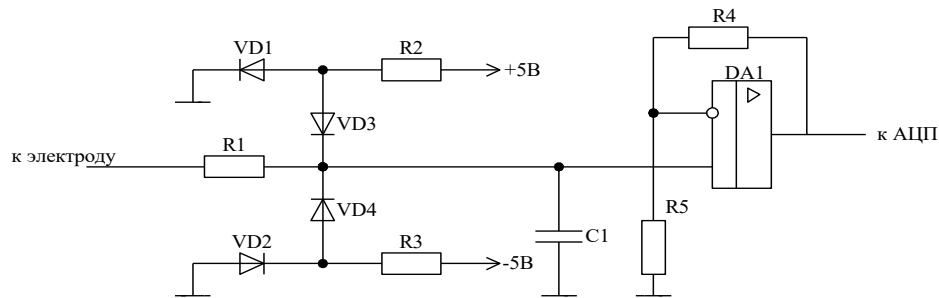


Рисунок 5.1 – Електрична схема підсилювача

Знаючи напругу живлення ланцюга R2-VD1 (5В) і зразкове падіння напруги на діоді VD1 (0.7В), знаходимо падіння напруги на резисторі R2:

$$U_{R2} = U_n - U_o \quad (5.7)$$

$$U_{R2} = 5 - 0.7 = 4.3В.$$

Задаємося струмом для прямого зміщення діодів рівним 1 мА. Отже, опір резистора R2, визначаємо за формулою

$$R2 = \frac{U_{R2}}{I}, \quad (5.8)$$

$$R2 = \frac{4.3}{1 \cdot 10^{-3}} = 4.3 кОм$$

Вибираємо резистор R2 типу P1-12-0.125-4.3кОм±5%.

Імпульс дефібрилятора має амплітуду 6кВ, тому перед доданими обмежувачем необхідно встановити додаткові засоби захисту від високої напруги. Як пристрій захисту приймаємо захисний стабілітрон типу 6РЕ100 з напругою пробою 100В, отже, напруга на вході ДО не перевищує 100В і струм через діоди ДО не перевищує  $I_{d\max}$ .

$$I_{d\max} \approx \frac{U_{вх\max}}{R1}; \quad (5.9)$$

де  $U_{вх\max}$  - максимальна напруга на вході ДО.

$$I_{d\max} \approx \frac{100}{10 \cdot 10^3} = 0.01 \text{ А}$$

Обираємо діод з урахуванням умов  $I_{пр} > I_{d\max}$ ,  $U_{об} > 2U_{жив}$ . Таким чином, вибираємо діод типу 1N914.

Обираємо операційний підсилювач DA1 типу OP200GP. Його параметри:

Напруга живлення, В	Uжив	5...20
Напруга живлення номінальна, В	Uжив ном	9
Струм, мкА	Iпот	570
Коефіцієнт посилення	KU	32·105
Вхідний струм, нА	Iвх	5
Вхідний струм зміщення, нА	dIвх	3.5
Вхідний опір:		
синфазний режим, Гом	Rвх синф	125
диференційний режим, Мом	Rвх диф	10
Опір навантаження, кОм	Rн	2

Максимальна вихідна напруга, В	U <sub>вих</sub> max	12.2
Струм шуму, пА ( $\Delta F = 0.01 \dots 100$ Гц)	I <sub>ш</sub>	15
Напруга шуму, мкВ ( $\Delta F = 0.01 \dots 100$ Гц)	U <sub>ш</sub>	0.5
Коефіцієнт придушення синфазного напруги, дБ	Кос сф	130
Коефіцієнт ослаблення перешкод напруги харчування, мкВ / В	Кос пит	$1.67 \cdot 10^6$
Температурний дрейф нуля, мкВ / °С	ТСОУ	0,5

Обираємо резистор R5 типу P1-12-0.125-10 кОм $\pm$ 5%.

Використовуючи формулу 1.1, розраховуємо опір резистора R4:

$$R4 = 10^3 \cdot (8 - 1) = 70 \text{ кОм}$$

Обираємо резистор R4 типу P1-12-0.125-68 кОм $\pm$ 5%.

У схемі електрокардіографа використовуються дев'ять вхідних підсилювачів, для побудови яких застосовуємо п'ять ОУ типу OP200PG.

## 5.2 Розрахунок і вибір елементів пристрою гальванічної розв'язки

Швидкість передачі даних по послідовному каналу зв'язку між АЦП і DSP становить 8 Мб / с при цьому частота проходження імпульсів в послідовному каналі зв'язку може досягати 8 МГц. Виходячи з вище сказаного оптрони УГР повинні передавати без спотворень прямокутні імпульси з частотою 8 МГц, і ізоляція між входом і виходом оптронів повинна витримувати напругу 4 кВ.

Вибираємо оптрон типу HCPL-4661 # 520, його параметри:

					ДП. ПІ 51.00.000 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дат		



Опір вхід-вихід, Ом	RI-O	1012
Максимальна напруга вхід-вихід, В	UI-O max	5000
Ємність входу, пФ	CI	60
Пряме напруга на вході, В	UI пр	1,4
Прямий вхідний струм, мА	Iвх	7,5
Прямий максимальний вхідний струм, мА	Iвх max	20
Зворотна напруга на вході, В	UI обр	5
Напруга живлення (вихідний частини), В	UO пит	5
Максимальна напруга на виході, В	UO max	7
Максимальний вихідний струм, мА	IO max	50
Максимальний час затримки сигналу, нс	tз max	75
Кількість оптопар, шт		2

Таблиця 5.1 – Параметри оптрону HCPL-4661 # 520

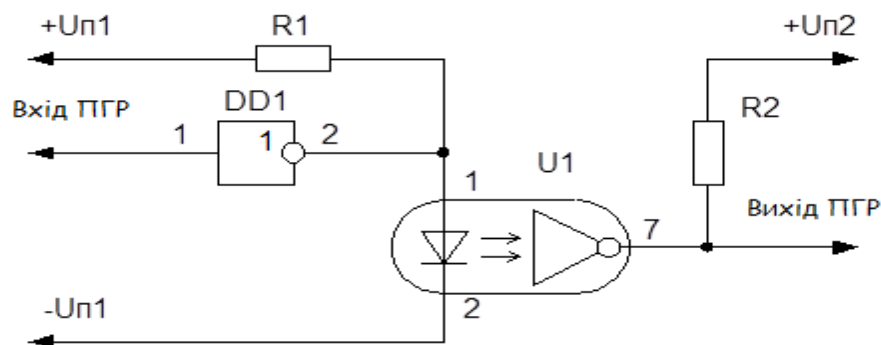


Рисунок 5.2 – Схема електрична принципова одного каналу пристрою гальванічної розв'язки

Для управління мікросхемами АЦП необхідна гальванічна розв'язка восьми цифрових сигналів, з них шість сигналів передаються в АЦП і два з АЦП. Гальванічна розв'язка всіх восьми сигналів здійснюється однаковими

схемами, одна з яких наведена на малюнку 1.8. Дана схема рекомендується технічною документацією на оптрон HCPL-4661 # 520. Мікросхема DD1 - шість інверторів з відкритим колектором на виході. Значення опору резистора R2 також рекомендуються технічною документацією на оптрон, і дорівнює 1 кОм. Вибираємо резистор R2 типу: P1-12-0.125-1 кОм.

Опір резистора R1 можна розрахувати за формулою:

$$R1 = \frac{U_{n1} - U_{Inp}}{I_{ex}} \quad (5.10)$$

де  $U_{ж1}$  - напруга живлення вхідної частини однакову 5В.

$$R1 = \frac{5 - 1,4}{7,5 \cdot 10^{-3}} = 480 \text{ Ом};$$

Вибираємо резистор R1 типу: P1-12-0.125-470 Ом.

### 5.3 Розрахунок похибки вимірювання амплітуди кардіосигналу

Розрізняють два види похибки вимірювання амплітуди сигналу: аддитивну і мультиплікативну. Аддитивна похибка аналого-цифрової частини розробленого електрокардіографа не впливає на точність вимірювання амплітуди кардіосигналу, так як всі вимірювання амплітуди виробляються щодо ізолінії. Ізолінія виділяється з кардіосигналу за допомогою цифрового фільтра низьких частот реалізованого програмно на ПЕОМ. З вищесказаного випливає що, залишається тільки мультиплікативна похибка вимірювання сигналу. Але необхідно врахувати помилки вимірювання пов'язані з аналого-цифровим перетворенням сигналу. Такі як, помилки при дискретизації сигналу по амплітуді, нелінійність характеристики АЦП, шуми аналого-цифрової частини схеми (шуми зменшують динамічний діапазон АЦП, а отже збільшують помилку дискретизації).

## 5.4 Розрахунок похибки аналого-цифрової частини ПММК

Для розробленої аналого-цифрової частини максимальна похибка буде складатися з температурних дрейфів джерела опорної напруги і АЦП, а також інтегральної нелінійності АЦП. А відхилення коефіцієнта передачі вхідних підсилювачів (обумовлені розкидом параметрів елементів) і абсолютна похибка в кінці шкали АЦП (зумовлена розкидом параметрів мікросхем АЦП) коригуються в мікропроцесорної частини, шляхом множення значення отриманого з АЦП на коефіцієнт обернено пропорційний відхиленню зазначених параметрів. Точність корекції визначається кількістю двійкових розрядів коригуючого коефіцієнта, і для тридцяти двох розрядного коефіцієнта становить:  $2,3 \cdot 10^{-8}\%$ , тобто залишковою похибкою (обумовленої розкидом параметрів елементів) можна знехтувати. Залишаються тільки складові залежать від температури і нелінійність АЦП. Тоді загальну похибку можна записати у вигляді:

$$\gamma = \frac{\Delta T \cdot (TC_{АЦП} + TC_{ВЛУ})}{U_{вхmax} - U_{вхmin}} \cdot 100\% \quad (5.11)$$

$\Delta T$  - максимальне відхилення навколишньої температури від нормальної;

$TC_{АЦП}$  - відхилення шкали АЦП при зміні температури на один  $^{\circ}\text{C}$ ;

$TC_{ІОН}$  - температурний дрейф джерела опорного напруги при зміні температури на один  $^{\circ}\text{C}$ .

$$\gamma = \frac{25 \cdot (30 \cdot 10^{-6} + 20 \cdot 10^{-6})}{2,5 - (-2,5)} \cdot 100\% = 0,025 \%$$

## 5.5 Розрахунок напруги шуму аналого-цифрової частини ПММК

Для розрахунку динамічного діапазону аналого-цифрової частини ПММК, проведемо розрахунок шумів. Динамічний діапазон визначається

					ДП. ПІ 51.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		35

максимальним рівнем вхідного напруги і рівнем шумів аналого-цифрової частини ПММК. Рівень шумів аналого-цифрової частини не повинен значно (в межах одного порядку) перевищувати напругу, відповідне одному молодшого розряду АЦП. Це виражається наступною формулою:

$$\frac{U_{\text{вхmax}} - U_{\text{вхmin}}}{2^N} \approx U'_{\text{ш об}} \quad (5.12)$$

де  $U_{\text{вхmax}}$  - максимальна вхідна напруга;

$U_{\text{вхmin}}$  - мінімальна вхідна напруга;

$N$  - кількість розрядів АЦП

$U'_{\text{ш про}}$  - загальну напругу шумів різних джерел, приведених до входу АЦП.

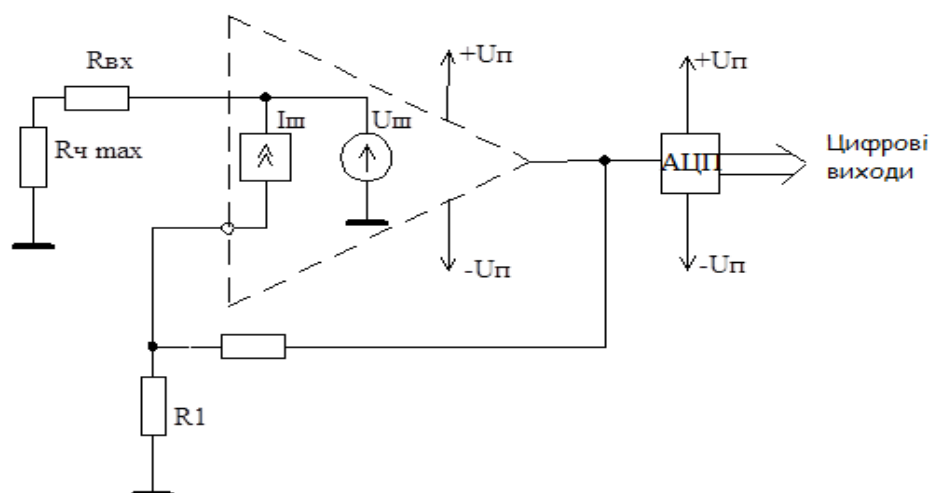


Рисунок 5.3 - Еквівалентна схема для розрахунку шумів в аналого-цифрової частини

Так як рівняння має наближене рівність, можна враховувати в розрахунку тільки найбільш вагомні джерела шуму, такі як, внутрішні шуми операційних підсилювачів і внутрішні шуми АЦП. Також в цій формулі необхідно врахувати вплив пульсацій напруги живлення, так як джерело харчування

розроблявся окремо від аналого-цифрового і мікропроцесорного блоку. Тому характер пульсацій напруги живлення можна вважати випадковим (шумоподібним). Таким чином, У'ш про аналого-цифровий і мікропроцесорної частини можна записати в наступному вигляді:

$$U'_{ш об} = \sqrt{\sum_{i=1}^n (U'_{ш i})^2} = \sqrt{(U_{ш ОУ} \cdot K_Y)^2 + (I_{ш ОУ} \cdot R_{\text{э}} \cdot K_Y)^2 + 2 \cdot \left( \frac{\Delta U_{П ОУ}}{K_{О ОУ}} \right)^2 + U_{ш АЦП}^2 + 2 \cdot \left( \frac{\Delta U_{П АЦП}}{K_{О АЦП}} \right)^2} \quad (5.13)$$

Де  $U_{ш}$  - напруга шуму операційного підсилювача;

$I_{ш}$  - струм шуму операційного підсилювача;

$R_e$  - еквівалентний опір зовнішніх ланцюгів підключених до операційного підсилювача;

$\Delta U_{П ОУ}$  - напруга пульсації джерела живлення операційного підсилювача;

$K_O$  - коефіцієнт ослаблення пульсації джерела живлення;

$U_{ш АЦП}$  - напруга шуму АЦП;

$\Delta U_{П АЦП}$  - напруга пульсації джерела живлення АЦП;

Еквівалентний опір зовнішніх ланцюгів, що підключаються до входів операційного підсилювача, розраховується за такою формулою:

$$R_{\text{э}} = \left( R_{ч \max} + R_{ex} + \frac{R1 \cdot R2}{R1 + R2} \right), \quad (5.14)$$

де  $R_{ч \max}$  - максимальний опір тіла людини;

$R_{вх}$  - вхідний опір операційного підсилювача;

$R1, R2$  - опору зворотного зв'язку операційного підсилювача.

$$R_3 = \left( 20 + 10 + \frac{10 \cdot 82}{10 + 82} \right) = 38.9 \text{ кОм}.$$

Коефіцієнт посилення з негативним зворотним зв'язком, що складається з опорів R1 і R2 приблизно дорівнює восьми.

Коефіцієнт ослаблення пульсацій напруги живлення дорівнює  $1.67 \cdot 10^6$

Напруга пульсації джерела живлення операційного підсилювача складає 20 мВ, відповідно до параметрів джерела живлення.

Напруга внутрішнього шуму АЦП становить 11 мкВ.

Коефіцієнт ослаблення напруження пульсації джерела живлення АЦП визначаємо за формулою:

$$K_0 = 20 \cdot \lg(U_{\text{ПАЦП}}) = 70 [\text{дБ}] \quad (5.15)$$

$$K_0 = 10^{\frac{70}{20}} = 3162.28 [\text{о.е.}]$$

Наведені вище параметри підставляємо в формулу для розрахунку загальної напруги шумів.

$$U'_{\text{ш об}} = \sqrt{(0.5 \cdot 10^{-6} \cdot 8)^2 + (15 \cdot 10^{-12} \cdot 38.9 \cdot 10^3 \cdot 8)^2 + 2 \cdot \left( \frac{20 \cdot 10^{-3}}{16.666 \cdot 10^5} \right)^2 + (11 \cdot 10^{-6})^2 + 2 \cdot \left( \frac{0.0063 \cdot 10^{-3}}{16.666 \cdot 10^5} \right)^2} = 16.6 \text{ мкВ}$$

Розрахуємо напругу, відповідну напрузі зміни молодшого розряду АЦП.

$$\Delta U = \frac{U_{\text{exmax}} - U_{\text{exmin}}}{2^N}, \quad (5.16)$$

де  $U_{\text{д}}$  - діапазон вхідної напруги АЦП;

N - кількість розрядів АЦП.

					ДП. ПІ 51.00.000 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дат		

$$\Delta U = \frac{2,5 - (-2,5)}{2^{22}} = 12 \text{ мкВ}.$$

З даного розрахунку видно, що вибір АЦП проведений правильно, і динамічний діапазон аналого-цифрової частини становить:

$$D_{AC} = 20 \cdot \lg \left( \frac{U_{\text{вхmax}} - U_{\text{вхmin}}}{U'_{\text{ш об}}} \right) \quad (5.17)$$

$$D_{AC} = 20 \cdot \lg \left( \frac{2,5 - (-2,5)}{16,6 \cdot 10^{-6}} \right) = 109,5 \text{ дБ}$$

Аналізуючи наведені розрахунки можна визначити загальну похибку при вимірюванні амплітуди кардіосигналу, по наведеній нижче формулі:

$$\gamma_{об} = \gamma + \frac{U_{шоб}}{U_{\text{вхmax}} - U_{\text{вхmin}}} \cdot 100\% + \delta_{ил} \quad ;(5.18)$$

$$\gamma_{об} = 0,025 + \frac{16,6 \cdot 10^{-6}}{2,5 - (-2,5)} \cdot 100\% + 0,003 = 0,028 \%$$

Загальна похибка вимірювання амплітуди кардіосигналу становить 0,028%, що значно менше, ніж у існуючих аналогів і задовольняють вимогам технічного завдання.

Розрахунок похибки вимірювання часових інтервалів кардіосигналу

На точність вимірювання інтервалів часу кардіосигналу впливає період дискретизації, точність вимірювання тимчасового інтервалу становитиме два періоду дискретизації. При частоті дискретизації дорівнює 1 кГц похибка вимірювання буде складати 2 мс, що значно менше ніж потрібно за технічним завданням.

Елемент	Позначення	Номінальна інтенсивність відмови $\lambda_0 \cdot 10^{-6}$ , 1/год	Кількість , шт
Резистори: RES0805	$\lambda_0$ R1	0,087	26
Конденсатори: CAP0805	$\lambda_0$ C1	0,04	45
Мікросхеми	$\lambda_0$ ІМС	0,013	15
Печатна плата	$\lambda_0$ ПП	0,7	1
Паяльне з'єднання	$\lambda_0$ ОМ	0,01	300

Таблиця 5.2 - Інтенсивність відмов за типами елементів

### Розрахунок інтенсивності відмов

Мікросхеми:

$$\lambda_0 = 0,013 \cdot 10^{-6} \cdot 15 = 0,195 \cdot 10^{-6} \text{ 1/год.}$$

Конденсатори:

$$\lambda_0 = 0,04 \cdot 10^{-6} \cdot 45 = 1,8 \cdot 10^{-6} \text{ 1/год.}$$

Резистори:

$$\lambda_0 = 0,087 \cdot 10^{-6} \cdot 26 = 2,26 \cdot 10^{-6} \text{ 1/год.}$$

Плата:

$$\lambda_0 = 0,7 \cdot 10^{-6} \cdot 1 = 0,7 \cdot 10^{-6} \text{ 1/год.}$$

Пайка:

$$\lambda_0 = 0,01 \cdot 10^{-6} \cdot 300 = 3 \cdot 10^{-6} \text{ 1/год.}$$

Інтенсивність відмови системи:

$$\lambda = 0,195 \cdot 10^{-6} + 1,8 \cdot 10^{-6} + 2,26 \cdot 10^{-6} + 0,7 \cdot 10^{-6} + 3 \cdot 10^{-6} = 7,955 \cdot 10^{-6} \text{ 1/год}$$



Середній час напрацювання на відмову:

$$T_{\text{ср.расч.}} = 1/\lambda = 1/7,955 = 125\,707 \text{ год};$$

$$T_{\text{ср.расч.}} > T_{\text{ср.}} \text{ (5.19)}$$

Середній час напрацювання на відмову у персональних кардіографів становить 5 років (43800 год). Отже, кардіограф задовольняє потребам.

## 6 ВИКОРИСТОВУВАНІ МОВИ ПРОГРАМУВАННЯ І ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

Додаток має бути розроблений для платформи Windows Phone 8.1 і має зворотну сумісність з Windows 10 Mobile. Перш ніж розглядати інструменти для Windows Phone 8.1, необхідно ввести деякі загальні поняття, з якими доведеться зіткнутися на даному етапі. Перше базове поняття - це SDK або Software Development Kit. Під ним розуміють набір засобів розробки, що дозволяє програмістам створювати додатки під деяку архітектуру - для певного пакету програм, апаратної або програмної платформи, операційної або комп'ютерної систем. SDK, як правило, надає розробнику широкий набір інструментів, який може включати в себе засоби налагодження і довідкові матеріали. Розробка додатків для платформи Windows Phone 8.1 пов'язана з групою інструментів, які надаються набором Windows Phone 8.1 SDK. При реалізації програми застосовувався мову високого рівня C # і мову розмітки XAML. Для реалізації заповнення бази даних необхідно використовувати стандартну мову запитів SQL. Для організації діалогу користувача і системи повинен застосовуватися графічний користувацький інтерфейс, який відповідає стандартам мобільної платформи Windows Phone. Як середовище розробки використовується Visual Studio 2015. І, так як додаток буде взаємодіяти з кардіографом за

					ДП. ПІ 51.00.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дат		

допомогою бездротового інтерфейсу передачі даних - Bluetooth, необхідно сказати пару слів і про цю технологію.

## 6.1 Мова програмування C #

C # є об'єктно-орієнтованою мовою, але підтримує також і компонентно-орієнтоване програмування. Розробка сучасних додатків все більше тяжіє до створення програмних компонентів у формі автономних і самоописних пакетів, що реалізують окремі функціональні можливості. Важлива особливість таких компонентів - це модель програмування на основі властивостей, методів і подій. Кожен компонент має атрибути, які надають декларативні відомості про компоненті, а також вбудовані елементи документації. C # надає мовні конструкції, безпосередньо підтримують таку концепцію роботи. Завдяки цьому C # відмінно підходить для створення і застосування програмних компонентів. [18]

Ось лише кілька функцій мови C #, що забезпечують надійність і стійкість додатків: прибирання сміття автоматично звільняє пам'ять, зайняту знищеними і невикористовуваними об'єктами; обробка виключень надає структурований і розширюваний спосіб виявляти і обробляти помилки; сувора типізація мови не дозволяє звертатися до неініціалізованих змінним, виходити за межі індексованих масивів або виконувати неконтрольоване приведення типів.

У C # існує єдина система типів. Всі типи C #, включаючи типи-примітиви, такі як int і double, успадковують від одного кореневого типу object. Таким чином, всі типи використовують загальний набір операцій, і значення будь-якого типу можна зберігати, передавати і обробляти схожим чином. Крім того, C # підтримує призначені для користувача посилальні типи і типи значень, дозволяючи як динамічно виділяти пам'ять для об'єктів, так і зберігати спрощені структури в стеці. [18]

					ДП. ПІ 51.00.000 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дат		

Щоб забезпечити сумісність програм і бібліотек C # при подальшому розвитку, при розробці C # багато уваги було приділено управлінню версіями. Багато мови програмування обходять увагою це питання, і в результаті програми на цих мовах ламаються частіше, ніж хотілося б, при виході нових версій залежних бібліотек. Питання управління версіями істотно вплинули на такі аспекти розробки C #, як роздільні модифікатори virtual і override, правила вирішення перевантаження методів і підтримка явного оголошення членів інтерфейсу. [18]

## 6.2 Опис мови XAML

XAML - це мова розмітки, який з'явився разом з першою версією WPF від Microsoft. Зараз він також використовується в Silverlight і Windows Phone 7 (сути той же Silverlight). Таким чином, зараз досить багато людей активно використовують XAML. Однак для ефективної роботи корисно буде зрозуміти концепції, які стоять за я мовою, щоб окремі конструкції не здавалися дивними. В першу чергу варто провести маленький ліричний відступ. Існує два основних види мов програмування: імперативні й декларативні. Імперативні мови - це всім відомі мови програмування, на зразок C, C ++, C #, Pascal, Basic і безліч інших. Основна ідея в тому, що в імперативній мовою ми говоримо, що потрібно зробити. Але не говоримо, що повинно вийти (зазвичай це ми повинні описати і перевірити в unit-тестах). Декларативні мови, в зворотну сторону, дозволяють нам описати стан, якого ми хочемо домогтися, але не вимагають (і зазвичай не дають) описати як прийти в цей стан. Приклади таких мов: XAML (та й взагалі все засновані на ієрархічній розмітці XML, HTML і т.п.), також SQL.

## 6.3 Середовище розробки Visual Studio 2015

Microsoft Visual Studio - лінійка продуктів компанії Майкрософт, що включають інтегроване середовище розробки програмного забезпечення і ряд інших інструментальних засобів. Дані продукти дозволяють розробляти

					ДП. ПІ 51.00.000 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дат		

як консольні додатки, так і додатки з графічним інтерфейсом, в тому числі з підтримкою технології Windows Forms, а також веб-сайти, веб-додатки, веб-служби як в рідному, так і в керованому кодах для всіх платформ, підтримуваних Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework і Microsoft Silverlight. [19]

Visual Studio включає в себе редактор вихідного коду з підтримкою технології IntelliSense і можливістю найпростішого рефакторінга коду. Вбудований відладчик може працювати як відладчик рівня вихідного коду, так і як відладчик машинного рівня. Решта вбудовуються інструменти включають в себе редактор форм для спрощення створення графічного інтерфейсу додатку, веб-редактор, дизайнер класів і дизайнер схеми бази даних. Visual Studio дозволяє створювати і підключати сторонні додатки (плагіни) для розширення функціональності практично на кожному рівні, включаючи додавання підтримки систем контролю версій вихідного коду (як наприклад, Subversion і Visual SourceSafe), додавання нових наборів інструментів (наприклад, для редагування і візуального проектування коду на предметно-орієнтованих мовах програмування або інструментів для інших аспектів циклу розробки програмного забезпечення (наприклад, клієнт Team Explorer для роботи з Team Foundation Server). [19]

## 6.4 Платформа Windows Phone 8.1

Платформа Windows Phone 8.1 належить до лінійки Windows NT, що означає, що вона заснована на одному ядрі з настільної операційної системою Windows 8.1. Це відкриває доступ мобільну операційну систему до потужностей, доступним звичайної настільної операційної системи. Ядро Windows NT прекрасно оптимізовано для роботи з багатоядерними процесорами, має можливість звернення до змінних носіїв і багато іншого.

					ДП. ПІ 51.00.000 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дат		

### 6.4.1 Архітектура платформи

На рисунку 6.1 зображена архітектура платформи Windows Phone 8.1.

Розглянемо докладніше, що означають компоненти архітектури.

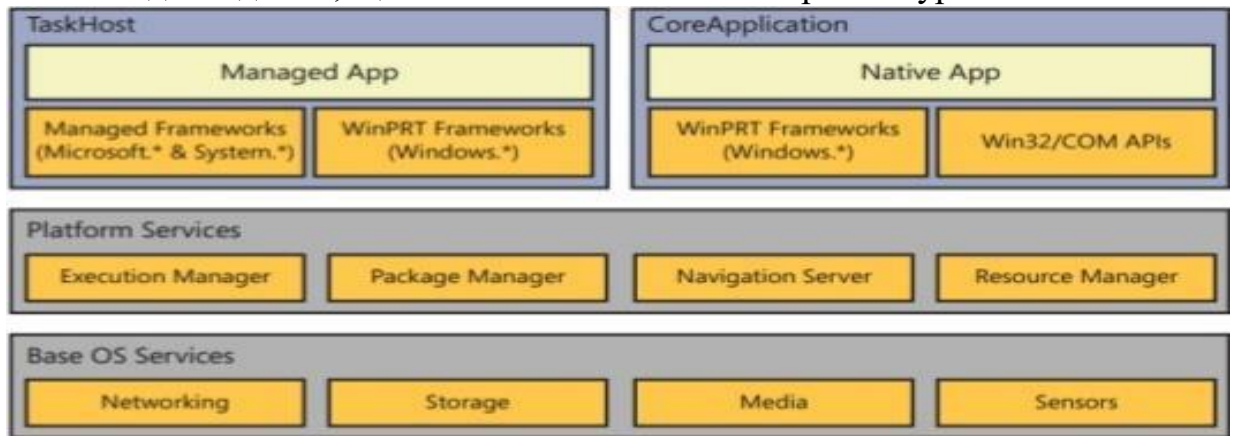


Рисунок 6.1, архітектура платформи Windows Phone

TaskHost і CoreApplication - дві різні моделі додатків. TaskHost - модель додатків, реалізованих за допомогою XAML розмітки. Цей підхід був основним ще з запуску першої версії платформи Windows Phone 7. CoreApplication представляє нову модель додатків на Windows Phone 8.1, яка заснована на моделі додатків для Windows 8. У випуску для Windows Phone 8.1 ця модель підтримує можливості нативних додатків з використанням Direct3D для призначеного для користувача інтерфейсу. Win32 / Com можуть бути використані в керованих додатках, а так само якщо вони обгорнуті в призначений для користувача компонент Windows Runtime. Ці дві моделі додатків базуються на основних сервісах платформи: Package Manager. Package Manager відповідає за установку і видалення програм, за збереження всіх метаданих додатків протягом усього періоду життя додатки. Він не тільки стежить за тим, які програми були встановлені і атестовані, а й зберігає інформацію про всі плитки додатків, які користувач закріпив на початковому екрані і про всі місця, де додаток може відображатися.

- Execution Manager

Execution Manager контролює всю логіку, пов'язану з виконанням програми протягом усього життєвого циклу програми. Він створює хостинг-процес для виконання програми та викликає події, пов'язані зі стартом / вимиканням / деактивацією додатки. Він виконує аналогічну задачу для фонових процесів додатки, а так само визначає належний порядок їх виконання.

- Navigation Server

Navigation Server управляє перемиканням між активними додатками на телефоні. Коли користувач натискає на іконку програми на стартовому екрані, він переходить зі стартового екрану до додатка, яке вибрав. Navigation Server відповідає за передачу наміри в Execution Manager, щоб обраної програми могло запуститися. Також, коли користувач натискає і утримує кнопку Назад і вибирає додаток, яке запускав до цього, Navigation Server повідомляє в Execution Manager, яка програма має бути реактивувати.

- Resources Manager

Resources Manager відповідає за забезпечення швидкості і чуйності інтерфейсу. Він стежить за використанням системних ресурсів усіма активними процесами, особливо за використання центрального процесора і пам'яті, і обмежує кількість цих процесів. Якщо додаток або фоновий процес перевищує відведений пул ресурсів, то він завершується. Все це лежить на вершині об'єднаного ядра Windows.

### 6.4.2 Безпека Windows Phone

Сучасні смартфони зберігають дуже багато особистої інформації користувача. Ця інформація повинна бути добре захищена. Модель безпеки

					ДП. ПІ 51.00.000 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дат		

Windows Phone заснована на моделі контейнерів безпеки - ізольовані контейнери, в яких процес створюється і виконується. Права доступу до контейнерів надаються системою. Система надає права за старим принципом найменших привілеїв, що означає, що програма не має доступу ні до чого, крім того, що необхідно для виконання поставлених функцій. Наприклад, додаток пошти не може довільно відкрити камеру, так як це не є його основною функцією. Кожен контейнер починається з обмеженого набору привілеїв, достатніх для написання самостійного застосування, такого як калькулятор чи проста гра, але недостатніх для використання всіх функцій смартфона. Якщо з додатком потрібно використовувати додаткові функції смартфона, такі як контакти користувача або позиціонування, ці функції повинні бути явно вказані в списку Можливостей, який зображенні на малюнку 2. Список Можливостей використовується як набір механізмів контролю доступу до функцій смартфона. Система повинна явно надати доступ до контейнера. При розробці програми розробник сам вказує функції, які йому необхідні для створення програми. Всі можливості смартфона, які необхідно використовувати в додатку, відображаються в магазині на сторінці додатка, і тоді користувач сам може вирішити, чи хоче він давати доступ вашого додатком до необхідних можливостей і встановити вашу програму чи ні.

### 6.4.3 Ядро

Як вже було сказано, платформа Windows Phone 8.1 отримала загальне ядро з настільної Windows 8.1. Насправді, ядро містить два розділених компонента. Перший компонент - Системне ядро Windows, яке включає в себе основні функції ОС Windows, в тому числі ядро NT, файлова система NT (NTFS) і мережевий стек. Це мінімальне ядро, яке є результатом вдосконалення архітектури протягом багатьох років, метою якого було

					ДП. ПІ 51.00.000 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дат		

забезпечити загальну базу для різних типів пристроїв, в тому числі і для смартфонів. Над системним ядром зібраний набір функцій Windows, які не входять в ядро, але так само доступні для смартфонів. Це такі компоненти, як Мультимедіа, CoreCRL, DirectX і Trident - движок рендеринга для Internet Explorer. Цей набір - Мобільний Ядро - це окремий архітектурний об'єкт для Windows Phone. Windows містить ті ж компоненти, що і мобільне ядро Windows Phone, але вони є частиною більш широкого набору функціональних можливостей. Це позначено пунктиром на рисунку 6.2.

Системне ядро і Мобільний ядро представляють об'єднання Windows і Windows Phone 8.1, коли дві операційні системи виконують один і той же код.

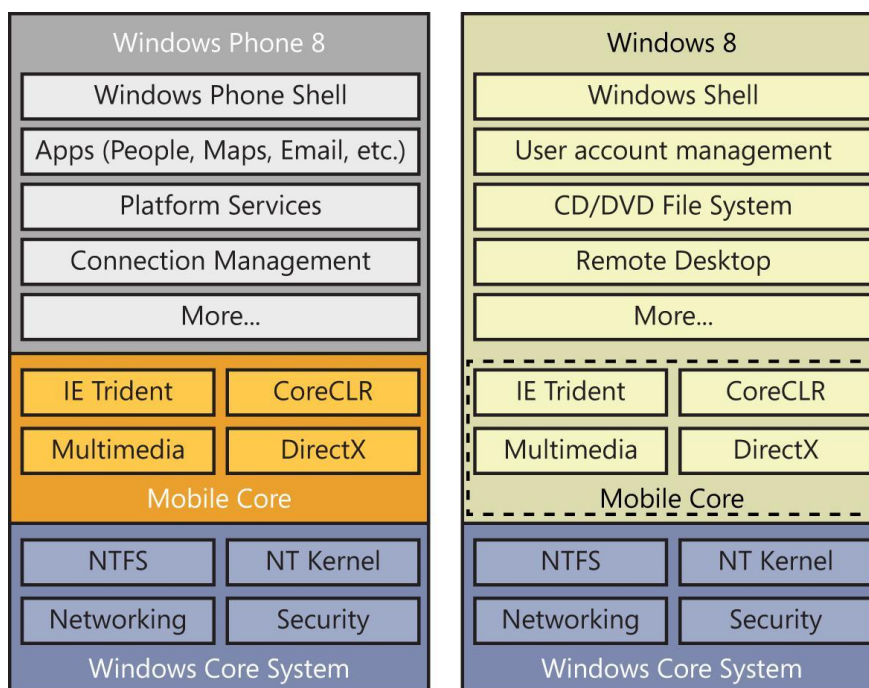


Рисунок 6.2, склад ядра Windows Phone 8, Windows 8

## 6.5 Інструменти для розробки

Для розробки додатків під Windows Phone 8.1, компанія Microsoft надала безкоштовний набір інструментів розробника Windows Phone 8.1 SDK. Він включає в себе [10].



- Microsoft Visual Studio 2015 для Windows Phone - середовище розробки програмного забезпечення.
- Microsoft Blend 2015 для Windows Phone - середовище для побудови інтерфейса користувача. Необходна для реалізації нетривіальних завдань дизайну користувацького інтерфейсу додатку.
- Емулятор пристрою на платформі Windows Phone 8 - необхідний для тестування додатків [10].
- Шаблони проектів, посилання на збірки, бібліотеки і заголовки бібліотек.
- Емулятор пристрою заснований на останній версії Microsoft Hyper-V, який вимагає 64-х бітний ЦПУ, який підтримує технологію SLAT (Second Level Address Translation - перетворення адрес другого рівня), технологія віртуалізації пам'яті, яку підтримувала більшість сучасних процесорів [10].

## 6.6 Інтерфейс користувача на платформі

Інтерфейс Windows Phone заснований на концепції мінімалізму. Майкрософт дотримується принципу: головне місце в призначеному для користувача інтерфейсі повинна займати важлива інформація, за якою користувач звернувся до пристрою. Всі інші елементи інтерфейсу або не потрібні, або повинні відійти на другий план. Основу користувацького інтерфейсу складають динамічні плитки (Tiles), які відображають динамічно мінливу важливу інформацію. На початку свого існування цей стиль називався Metro, але в 2012 році компанія Майкрософт прийняла рішення більше не використовувати цей термін. На даний момент, додатки, створені в цьому стилі, називають - додатки магазину Windows [10].

## 6.7 Структура проекту Windows Phone

Головним і найважливішим файлом проекту є WManifest.xml. Він містить всю важливу інформацію, яку операційна система повинна знати

					ДП. ПІ 51.00.000 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дат		

про програму [10]. Деякі пункти в WMAppManifest.xml, наприклад системні вимога, використовуються в процесі прийняття додатка в магазин Windows Phone. Маніфест містить у собі:

- Назва додатка;
- Іконки додатки для початкового екрану і для списку додатків;
- Підтримувані дозволи додатки;
- Список вимог до обладнання, яке необхідно додатком для виконання своїх функцій. Наприклад, наявність камери;

- Список можливостей, які необхідні додатком для виконання його функцій. Наприклад, доступ до галереї фотографій користувача. У Visual Studio 2015 доступний графічний інтерфейс для редагування Маніфесту додатки. Папка Assets (Активи) надана для зберігання зображень, необхідних для застосування. Файли Resources \ AppResources.resx і LocalizedStrings.cs забезпечують платформу для локалізації додатки. App.xaml забезпечує зручне розташування для зберігання ресурсів, які необхідні для застосування. Наприклад, стилі елементів управління. App.xaml.cs містить код, що забезпечує запуск програми, а так само обробники подій життєвого циклу програми. Для того щоб визначити події при відкритті або закритті програми, переходу додатки в паузу або відновлення програми, потрібно заповнити вже згенеровані обробники цих подій. Додаток Windows Phone складається зі сторінок, за якими відбувається навігація. Початкову сторінку можна так само вказати в Маніфесті додатки. За замовчуванням, початкова сторінка називається MainPage.xaml і генерується автоматично з загальним шаблоном сторінки додатка. Це основні складові кожної програми для Windows Phone. Структура може змінюватися в залежності від парадигми програмування, якою користується розробник.

					ДП. ПІ 51.00.000 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дат		

## 6.8 Життєвий цикл додатку

В життєвому циклі Windows Phone додатки існує три стану. У кожен момент часу додаток може бути, або Чи не запущено, або Запущено, або Припинено, що проілюстровано на рисунку 6.3.

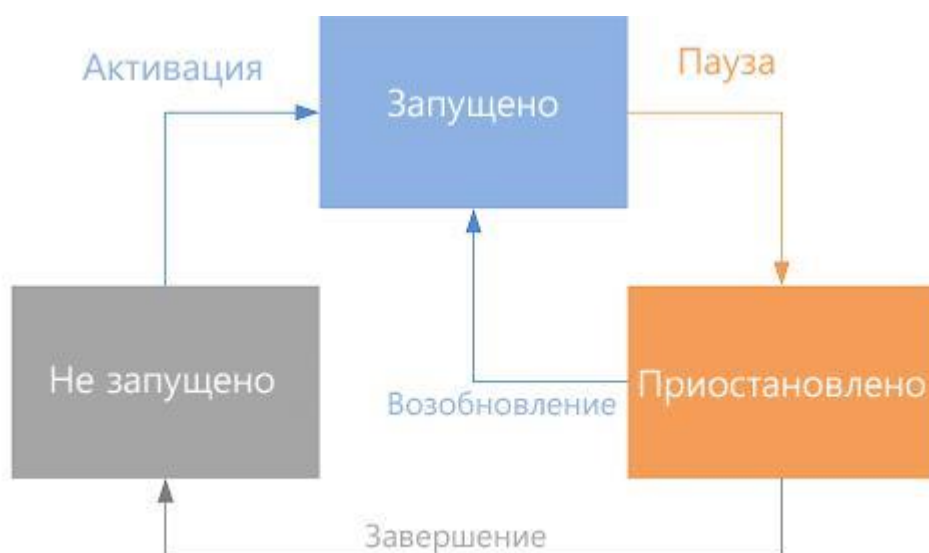


Рисунок 6.3, життєвий цикл програми.

## 7 BLUETOOTH

Bluetooth є стандартом бездротового радіозв'язку, що зв'язує на коротких відстанях різні типи пристроїв і призначеним для створення особистих мереж. Технологія Bluetooth використовує ту ж 2,4 ГГц робочу частоту, що і при використанні бездротових стандартів 802.11b і 802.11g, але, в свою чергу, під час роботи споживає меншу кількість енергії.

Низький рівень споживання енергії призводить до набагато більш вузькому діапазону і більш повільними темпами передачі даних, але цей спосіб з'єднання завоював свою нішу і використовується практично повсюдно.

Існують три класи радіоприймачів Bluetooth. Клас 1, який використовується в промислових цілях і забезпечує радіус дії до 100 метрів; при цьому для харчування використовується 100 мВт. Клас 2 є найбільш широко використовуваним радіоканалом, який використовує тільки 2,5 мВт і надає радіус дії до 10 метрів. Клас 3 використовує 1 мВт для харчування, але і радіус охоплення всього 1 метр.

### 7.1 Bluetooth low energy

Також необхідно розповісти про бездротова технології Bluetooth з низьким енергоспоживанням.

Принцип роботи BLE описаний вже в його назві: Low Energy. Протокол передбачає передачу даних короткими пакетами по необхідності, потім - вимкнення передавача. Низький рівень споживання енергії частково досягається застосуванням саме цього принципу. Замість класичного тандема в звичайному Bluetooth, пристрої BLE зв'язуються один з одним лише при необхідності відправки або отримання інформації. [20]

Протокол BLE строго структурований за принципом своєї комунікації з іншими пристроями. Спочатку девайси вивчають доступні сервіси для відправки / прийняття даних; невід'ємна частина цих сервісів - їх характеристики (characteristics), що визначають тип даних для майбутньої передачі. Характеристики, з міркувань наочності, можуть мати в своєму складі опису-дескриптори (descriptors), які допомагають визначити тип даних. Наприклад, розберемо сервіс під назвою «Heart Rate Monitor»

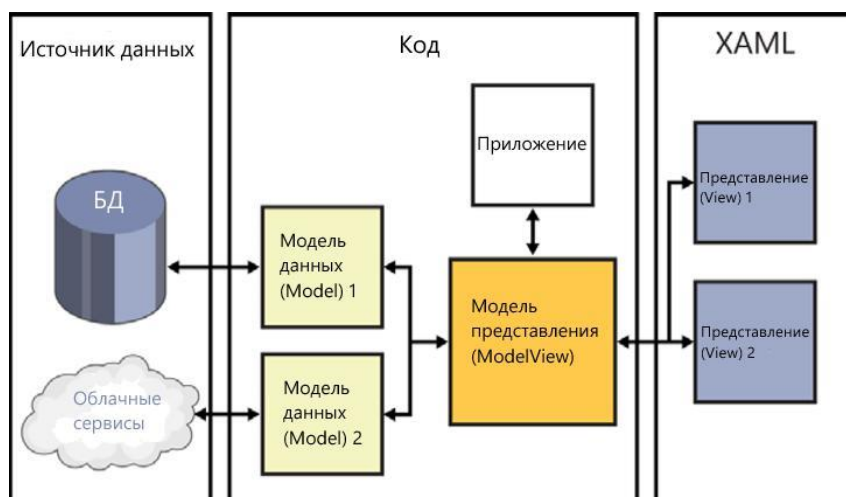
(монітор частоти серцебиття) - серед його характеристик присутні такі, як «вимір пульсу». Більшість API для Bluetooth LE дозволяють шукати локальні пристрої та визначати доступні в них сервіси, характеристики і дескриптори. [20]

## 7.2 Архітектура програми

При розробці використовувався шаблон проектування Model-View-ViewModel (MVVM) - патерн дуже часто використовується при написанні сучасних додатків, в тому числі Windows Phone додатків. Це еволюція шаблону Model-View-Controller (MVC). MVVM використовують для того, щоб відокремити код від призначеного для користувача інтерфейсу. Такий підхід полегшує паралельну роботу над кодом і дизайном додатки. Дизайнер працює в середовищі Expression Blend, а програміст паралельно пише код в Visual Studio. Це так само полегшує тестування. Розробник може писати тести незалежно від інших верств розробки. Існують три роздільні частини:

- View (візуальна частина) - це призначений для користувача інтерфейс, представлений кодом XAML і сторінками додатка;
- Model (модель даних) - це об'єкти даних, що представляють зв'язок з джерелом даних;
- ViewModel (Модель подання) - ця частина еквівалентна контролера в MVC, який виступає посередником між моделлю даних і поданням даних. Як правило, DataContext (Контекст даних) уявлення (View) пов'язаний з екземпляром моделі уявлення (ViewModel). А модель уявлення, як правило, пов'язана з екземпляром моделі даних (Model). У Windows Phone так само застосовуються Dependency Injection (DI - Впровадження залежності - процес надання зовнішньої залежності програмного коду). З DI, коли компонент залежить від іншого компонента, то це не є жорсткою

залежністю: замість цього, є списком сервісів, які необхідні одному компоненту від іншого. У Windows Phone DI використовується для забезпечення зв'язку уявлення (View), моделі уявлення (ModelView) і моделлю даних (Model), так що з додатком не потрібно безпосередньо пов'язувати їх. На малюнку 3.1 видно, що Подання, Модель даних і Модель кончини повністю розділені. З огляду на заснований на сторінках призначений для користувача інтерфейс Windows Phone додатків, дуже важливо відзначити, що можна використовувати одні і ті ж подання на різних сторінках. З цих причин, уявлення (сторінки) не впливають на створення Моделі уявлення. Швидше, додаток створює Модель уявлення та розкриває її як властивість, яка стає доступною з будь-якої сторінки. Підхід MVVM вітається в коді, і кілька шаблонів Visual Studio генеруються на його основі.



Малюнок 7.1. Огляд рівнів паттерна MVVM

## ВИСНОВКИ

1. У виконій роботі була обгрунтована актуальність розробки системи моніторингу пульсу людини.
2. Первинний прототип пристрою не призначений для широкого користування. Подальша мета роботи - упрощення і модернізація прототипу, робота над ергономічністю і естетичністю приладу.
3. У найближчому майбутньому потрібна розробка програмного забезпечення для автоматичної розшифровки ЕКГ. Створення тестової версії хмарного сервісу.
4. Надалі потрібні достовірні медичні дані, дослідження в лікарнях, для того щоб довести достовірність вимірювань. Після успішних випробувань можна приступати до написання бізнес-моделі проекту, залучення інвесторів і вихід на ринок.

					ДП. ПІ 51.00.000 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дат		

## ПЕРЕЛІК ПОСИЛАНЬ

1. Кардіограф на Arduino своїми руками. Режим доступу : <http://www.prointellekt.ru/EKG1.php> - Дата доступу : 10.05.2019
  2. Компьютерні методи автоматичного аналізу ЕКГ у системах кардіологічного спостереження. / Калиниченко, Александр Николаевич Режим доступу <http://www.dissercat.com/content/kompyuternye-metody-avtomaticheskogo-analiza-ekg-v-sistemakh-kardiologicheskogo-nablyudeniya> - Дата доступу 14.05.2019
  3. Порхунов А.А. АНАЛИЗ АКТУАЛЬНОСТИ И ПОДХОДОВ К СОЗДАНИЮ КАРМАННОГО ЭЛЕКТРОКАРДИОГРАФА // Материалы VI Международной студенческой электронной научной конференции «Студенческий научный форум Режим доступу <http://www.scienceforum.ru/2014/343/1114/> - Дата доступу 20.05.2019)
  4. А.Б.Сергиенко "Цифровая обработка сигналов" / Сергиенко А. Б. Питер, 2002. — 608 с
  5. Интерполирующие цифровые фильтры Режим доступа : <http://digteh.ru/digital/Intrpltr.php> - Дата доступу 01.06.2019
  6. jSSC (Java Simple Serial Connector) / Соколов Алексей – Режим доступу : <https://code.google.com/p/java-simple-serial-connector/> - Дата доступу 03.06.2019
  7. Arduino Language Reference Режим доступу: <http://www.arduino.cc/en/Reference/HomePage> - Дата доступу 07.02.2015
- Типові норми належності вогнегасників (затверджено наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 2



квітня 2004 р. N 151)

8. Джозеф Албахари, Бен Албахари C# 5.0. Справочник. Полное описание языка. :Пер. с англ. – М.: ООО “И.Д. Вильямс”, 2014. – 1008 с. :ил – Парал. тит. англ. ISBN 978-5-8459-1819-2, 978-1-4493-2010-2
9. Microsoft Corporation. Разработка приложений среды выполнения Windows. [Электронный ресурс]. URL: <https://msdn.microsoft.com/ru-ru/library/windows/apps/dn726537.aspx>
10. Whitechapel A. Windows Phone 8 Development Internals / S. McKenna, A. Whitechapel – North Sebastopol: «O`Reilly», 2013. – 965 с.
11. Xiao Mi band protocol analyze [Электронный ресурс]. changy-.github.io, 10.06.2019. – URL: <http://changy-.github.io/articles/xiao-mi-band-protocol-analyze.html>
12. Медицинская электронная аппаратура для здравоохранения – М.:2012 – 260с
13. Марков С. Цифровые сигнальные процессоры – М.: «Микроарт». 2011 – 138с
14. Паспорт. Электрокардиограф ЭК1Т-04 модель 017
15. Альбом рисунков и схем аппарата к паспорту. Электрокардиограф одноканальный с микропроцессорным управлением ЭК1ТЦ 01 модель 034
16. РЖ «Изобретения» 1996-2014г
17. Навчальні конструктори та DIY електроніка. [Електронний ресурс]: <http://hobbytech.com.ua/%D1%8D%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BF%D0%B8%D1%82%D0%B0%D0%BD%D0%B8%D0%B5-arduino/>
18. Короткий опис мови C# [Електронний ресурс]: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/>
19. Опис середовища Microsoft Visual Studio [Електронний ресурс]: [:https://dic.academic.ru/dic.nsf/ruwiki/56677](https://dic.academic.ru/dic.nsf/ruwiki/56677)
20. Опис Bluetooth Low Energy [Електронний ресурс]: <http://developer.alexanderklimov.ru/android/theory/ble.php>

					ДП. ПІ 51.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		57

Додатки

					ДП. ПІ 51.00.000 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дат		

## Додаток А

### Код програми в Arduino

```
#include <SPI.h>
#include <SD.h>
const int MAXSIZE=128; //максимальный размер буфера
int buf[MAXSIZE]; // создание массива
int bS; // сколько элементов в массиве уже есть
long tt, batDel;
File f;

void setup() {
    pinMode(A2,OUTPUT); //светодиод зелёный
    pinMode(A1,OUTPUT); //светодиод красный
    pinMode(A0,INPUT); //AD8232
    pinMode(10, OUTPUT); //PORTB 2 - SPI SD SS
    pinMode(11, OUTPUT); //PORTB 3 - SPI SD MOSI
    pinMode(12, INPUT); //PORTB 4 - SPI SD MISO
    pinMode(13, OUTPUT); //PORTB 5 - SPI SD SCK
    tt=0;
    bS=0;
    batDel=0;
    SD.begin(10);
    f=SD.open("ecg.txt",FILE_WRITE);
}

long readVcc(){
    long result;
    ADMUX = _BV(REFS0) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1); delay(2);
    ADCSRA |= _BV(ADSC);
    while (bit_is_set(ADCSRA,ADSC));
    result = ADCL;
    result |= ADCH<<8;
    result = (1.1*1023.0*1000) / result;
    return result;
}
```

					<i>ДП. ПІ 51.00.000 ПЗ</i>	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дат		

```

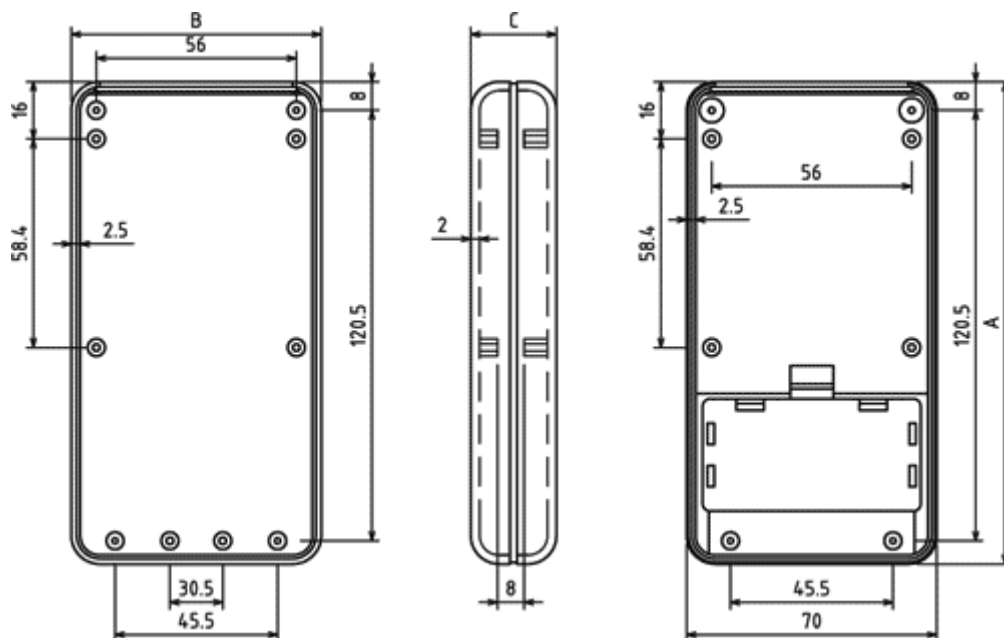
void loop() {
    buf[bS++]=analogRead(A0); //считываем значение, записываем в буфер и увеличиваем bS

    if(buf[bS-1]>=900) //если значение напряжения > или = 900, загорается зелёный светодиод
        digitalWrite(A2, HIGH);
    else
        digitalWrite(A2, LOW);
    if(batDel<=0){ //если время до мигания красного светодиода истекло
        batDel=(int)(500*readVcc()); //задаём время до следующего мигания красного светодиода
        //чем ниже ёмкость, тем меньше время, т.е. мигать светодиод начинает чаще
        digitalWrite(A1, HIGH); //зажигаем красный светодиод
    }
    if(bS==MAXSIZE){
        for(int i=0; i<bS; ++i, tt+=10){
            f.print(tt);
            f.print(' ');
            f.println(buf[i]); //выводим буфер в файл
        }
        bS=0;
        f.flush(); //выгрузка данных в файл, чтобы они не удалились
    }else
        delay(10);
    batDel-=10; //уменьшаем счётчик времени до следующего мигания
    digitalWrite(A1, LOW); //гасим красный светодиод
}

```

## Додаток Б

### Габаритні розміри корпусу



### Загальний вигляд приладу



					ДП. ПІ 51.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дат		61